Freeform Search

Database: US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term: L15 and weight$

Display: 50 **Documents in Display Format:** - **Starting with Number** 1

Generate: ○ **Hit List** ◉ **Hit Count** ○ **Side by Side** ○ **Image**

Search Clear Interrupt

Search History

DATE: Saturday, January 14, 2006 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L16	L15 and weight$	8	L16
L15	L14 and cluster	8	L15
L14	L13 and measurement	21	L14
L13	string near similarity	105	L13
L12	L11 and entity	24	L12
L11	L10 and match$	33	L11
L10	L9 and (difference)	35	L10
L9	L8 and (classif$ or cluster$)	44	L9
L8	L6 and (compar$)	82	L8
L7	L6 and (reference near tiems)	0	L7
L6	L5 and (current near items)	113	L6
L5	L2 and (items near list)	1698	L5
L4	L3 and (compar$ near list)	5	L4
L3	L2 and (reference near item$)	122	L3
L2	L1 and (item near list)	1698	L2
L1	707/$.ccls.	31789	L1

L4: Entry 4 of 5 File: USPT Apr 10, 2001

DOCUMENT-IDENTIFIER: US 6216140 B1
TITLE: Methodology for the efficient management of hierarchically organized information

Detailed Description Text (17):
In software development, while changes made to a hierarchy of files and directories generally need to be kept distinct, after a time it is often desirable to know what has changed since the original hierarchy or other baseline hierarchy. In order to accommodate this need, the present invention provides for the efficient comparison of a changed hierarchy with a baseline hierarchy. If two items reference the same database address, then they are shared. Because it is possible to know what is shared and what is not shared between two hierarchies being compared, the comparison between the two hierarchies, then, only encompasses those portions of the hierarchies that are not shared. Comparison of shared portions of the hierarchies is unnecessary and therefore inefficient because shared portions of the hierarchies are, by definition, equivalent software.

Detailed Description Text (19):
The efficient comparison operation of the present invention is based on the virtual copying and sharing operations of the hierarchies being compared, unique item identifiers that maintain item identity across all copies, and the change records contained within revision history 14. The comparison operation consists of instructions for pushing, or adding, both root directories of the hierarchies being compared to a list of differences between the hierarchies, instructions for merging items in this difference list and eliminating those that represent shared sub-hierarchies, and instructions for recursing appropriate, remaining items in the difference list. It is desirable in the comparison operation to minimize the recursion operation.

Current US Cross Reference Classification (1):
707/10

Current US Cross Reference Classification (2):
707/203

CLAIMS:

3. A method for the efficient comparison of two or more hierarchies of files and directories, comprising the steps of:

a) initializing a difference list by adding the root directory of a first hierarchy and the root directory of a second hierarchy to the difference list, wherein each item of the first hierarchy and each item of the second hierarchy has an identifier and a database address associated with it and wherein the first hierarchy is a first virtual copy of a hierarchy shared by a plurality of versions and was created by adding a first single link to a root of the hierarchy inclusive and wherein the second hierarchy is a second virtual copy of the hierarchy and was created by adding a second single link to the root of the hierarchy inclusive;

b) determining whether a first item of the first hierarchy has a first identifier that is equal to a second identifier of a second item of the second hierarchy;

c) merging the first item of the first hierarchy with the second item of the second hierarchy

L4: Entry 5 of 5 File: USPT Oct 20, 1998

DOCUMENT-IDENTIFIER: US 5826252 A
TITLE: System for managing multiple projects of similar type using dynamically updated global database

Detailed Description Text (7):
The common project management program is preferably written as a customized application of a conventional database program which will run on servers, PCs, and laptops alike, for example, MS Access.TM. 2.0 for Windows 3.1 or DOS 6.0 sold by Microsoft Corp., of Redmond, Wash. The program interface allows a user to select program databases and display lists which are stored in tables within the program databases. The lists can have pointers to project files and reference files which are stored in the uniform directory and file structure imported from the database. The program allows the user to create, add, edit, and delete items from the lists, and to generate, import, and export documents, reports, and other files. The program encompasses list management functions and database management functions.

Detailed Description Text (51):
FIG. 4 illustrates the basic data flow for a Revision Control procedure implemented in this example of the PATH program for a project services vendor. For the purpose of Revision Control, PATH is broken into components and each component is tracked by a separate version number. The major components being the PATH software, PATH Master Templates (which stores checklist items and reference files), Mission PATH Databases (which stores reference files), Pole PATH Databases (also stores reference files), and Processes & Tools (each Process or Tool is designated as a separate component). In the following description, numbers in parentheses are used in the diagram to reference the sequence of steps of the Revision Control procedure.

Current US Original Classification (1):
707/1

Current US Cross Reference Classification (4):
707/102

Current US Cross Reference Classification (5):
707/2

Current US Cross Reference Classification (6):
707/201

CLAIMS:

9. The system as described in claim 8, wherein said Combine means includes means for creating a combined list of best current data items by first copying best current data items for the participating project to the combined list, then comparing each new best current data item imported from the global project management database for inclusion on the combined list.

19. The method as described in claim 18, wherein said combining step includes creating a combined list of best current data items by first copying best current data items for the participating project to the combined list, then comparing each new best current data item imported from the global project management database for inclusion on the combined list.

into a single entry in the difference list if the first identifier of the first item is equal to the second identifier of the second item of the second hierarchy indicating that the first item and the second item are shared;

d) repeating steps b and c for each item of the plurality of items of the first hierarchy;

e) determining whether the first item of the first hierarchy has a first database address that is equal to a second database address of the second item of the second hierarchy;

f) removing the single entry from the difference list if the first database address of the first item of the first hierarchy is equal to the second database address of the second item of the second hierarchy;

g) repeating steps e and f for each item of the plurality of items of the first hierarchy so that the difference list includes only items that are not shared by the first hierarchy and the second hierarchy; and

h) recursively expanding the difference list by adding a plurality of sub-items onto the difference list for each of the items on the difference list and repeating steps b through h as required so that the difference list contains all differences between the plurality of items of the first hierarchy and the plurality of items of the second hierarchy,

wherein the time required for a computer to perform a) to h) is determined by a number of differences between the first and second hierarchies and not by a size of the first and second hierarchies.

12. A computer readable storage media containing a computer program for the efficient comparison of two or more hierarchies of files and directories, comprising:

a) instructions for initializing a difference list by adding the root directory of a first hierarchy and the root directory of a second hierarchy to the difference list, wherein each item of the first hierarchy and each item of the second hierarchy has an identifier and a database address associated with it and wherein the first hierarchy is a first virtual copy of a hierarchy shared by a plurality of versions and was created by adding a first single link to a root of the hierarchy inclusive and wherein the second hierarchy is a second virtual copy of the hierarchy and was created by adding a second single link to the root of the hierarchy inclusive;

b) instructions for determining whether a first item of the first hierarchy has a first identifier that is equal to a second identifier of a second item of the second hierarchy;

c) instructions for merging the first item of the first hierarchy with the second item of the second hierarchy into a single entry in the difference list if the first identifier of the first item is equal to the second identifier of the second item of the second hierarchy indicating that the first item and the second item are shared;

d) instructions for repeating instructions b and c for each item of the plurality of items of the first hierarchy;

e) instructions for determining whether the first item of the first hierarchy has a first database address that is equal to a second database address of the second item of the second hierarchy;

f) instructions for removing the single entry from the difference list if the first database address of the first item of the first hierarchy is equal to the second database address of the second item of the second hierarchy;

g) instructions for repeating instructions e and f for each item of the plurality of items of the first hierarchy so that the difference list includes only items that are not shared by the first hierarchy and the second hierarchy; and

h) instructions for recursively expanding the difference list by adding a plurality of sub-items onto the difference list for each of the items on the difference list and repeating instructions b through h as required so that the difference list contains all differences between the plurality of items of the first hierarchy and the plurality of items of the second hierarchy.

Previous Doc Next Doc Go to Doc#